“LAB 4- CROSS-VALIDATION USING K-FOLD METHOD”

IENG3304 – DATA MANAGEMENT & ANALYTICS

MAY 31, 2022

WRITTEN BY: ABILASH SURENDRAN
BANNER ID: B00891410

<u>Table Of Content:</u>

<u>LIST OF FIGURES:</u>

Introduction:

Machine learning is a new data-analysis technique that is still in its infancy. Everything in today's world is data-driven. Every sector in the global market requires data analysis and demand forecasting in order to remain competitive. Regression and classification models come in handy in these situations. Several regression and classification models were constructed in prior lab tasks. However, in this lab, the earlier models will be validated using a resampling technique known as k fold cross-validation. The k fold value can be set to 5, 10, or 20, and the relevant error bar plots can be found. Their average square errors are calculated as well, and the optimal model for both regression and classification is determined.

The regression dataset is the Spanish wine dataset, which has 7401 rows and 11 columns. The data demonstrates how various variables influence the wine's quality. A heart dataset serves as the basis for the classification model. The patient's age, sex, type of chest pain, resting blood pressure, cholesterol, fasting blood sugar, resting electrocardiogram, maximum heart rate, exercise-induced angina, old peak, and heart disease, which is the output parameter, are all included in the dataset, which has 918 rows and 12 columns.

BODY:

CLASSIFICATION:

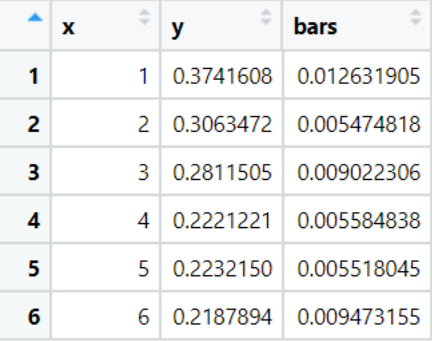
Classification is a process in r script which can be performed to predict a categorical label of a data object based on its features and properties. With the heart dataset, the data was completely cleaned and analysed in lab week 3. Moreover, several classification models were formulated. These classification models are then incorporated with k fold cross-validation to analyse the feasibility of the model. Here the value of k is 5. Therefore, the cross-validation model can also be called a 5-fold cross-validation.

In r script, classification is a procedure that predicts a categorical label for a data object based on its features and properties. In lab assignment 3, the data from the heart dataset was entirely cleaned and analysed. In addition, numerous classification models were developed. These classification models are then tested using k fold cross-validation to see if they are feasible. The value of k is 5 in this case. As a result, the cross-validation model is often known as a 5-fold cross-validation model. The six different models that I developed to determine whether or not a person has heart disease are listed below:

The first model is built by considering three predictors: cholesterol, resting blood pressure and fasting blood sugar. The second model involves predicting whether a person will be diagnosed with heart disease based on factors such as age and maximal heart rate. The third model is constructed by simply adding fasting blood sugar to the second model. The fourth model includes four predictors: age, blood sugar, fasting blood sugar, and chest pain type. Five predictors are included in the fifth model: age, blood sugar, fasting blood sugar, kind of chest discomfort, and resting ECG. The final model, also known as the sixth model, includes factors such as age, sex, old peak, and exercise angina.

Because the dataset lacks significant individual variables that influence the outcome, I did not utilise single predictors to classify whether an individual will have heart disease or not. As a result, numerous predictors are utilised in a model to better identify the outcome. To find the best model, all six classification models are developed and tested using cross-validation. Several iterations were carried out, with the findings stated below.

Cross-validation for Classification model attempt 1:



	x	y	bars
1	1	0.3741608	0.012631905
2	2	0.3063472	0.005474818
3	3	0.2811505	0.009022306
4	4	0.2221221	0.005584838
5	5	0.2232150	0.005518045
6	6	0.2187894	0.009473155

Here x describes the model number, y depicts the average square error and bars depict the variance. The graph below shows the standard error rate and model complexity. The error bars are used to depict the error range. Every model has a variance range which extends as a vertical bar. It is clear from the plot that model 4 has the lowest standard error. That is, it has a standard error of 22% which is the lowest of every other model. However, model 5 still falls within the variance of the fourth model. However, we cannot opt for model 5 because of the increase in model complexity. That is model 5 consists of 5 predictors while model 4 has only four predictors. In this instance, model 4 gives reliable outcome than the other five models.

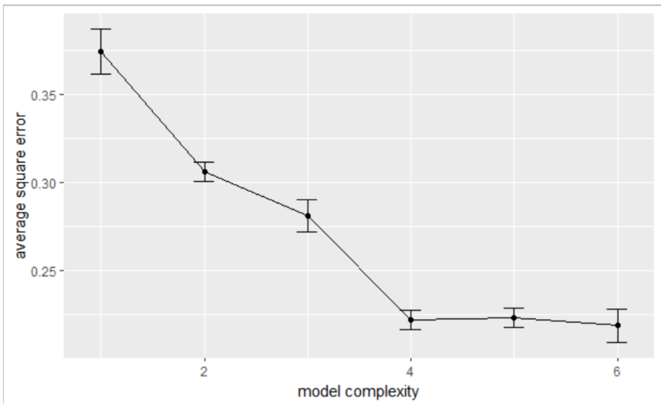
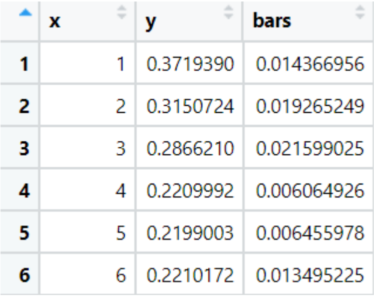


Fig 1: Plot Consisting of Error Bars Which is Spread Across Six Models with first randomised test subset

Cross-validation for Classification model attempt 2:



	x	y	bars
1	1	0.3719390	0.014366956
2	2	0.3150724	0.019265249
3	3	0.2866210	0.021599025
4	4	0.2209992	0.006064926
5	5	0.2199003	0.006455978
6	6	0.2210172	0.013495225

Here x describes the model number, y depicts the average square error and bars depict the variance. The graph below shows the average square error rate and model complexity. The error bars are used to depict the error range. Every model has a variance range which extends as a vertical bar. It is evident from fig 2, that there is a visible difference in variance for model 3 and model 6 with the respect to that of the previous attempt. However, model 4 has an average square error rate of 22%, while model five has an average square error rate of 21% Also, model 5 is within the variance of the fourth model. However, we cannot opt for model 5 because of the increase in model complexity. Moreover, model six still has only 4 predictors and gives the standard error value to be 22% and has comparatively more variance than model 4. On a practical basis, it is easier to predict heart disease with age, blood sugar, fasting blood sugar and the type of chest pain than with predictors like exercise angina and old peaks. Therefore, on a practical basis still, model 4 seems to be the best model.

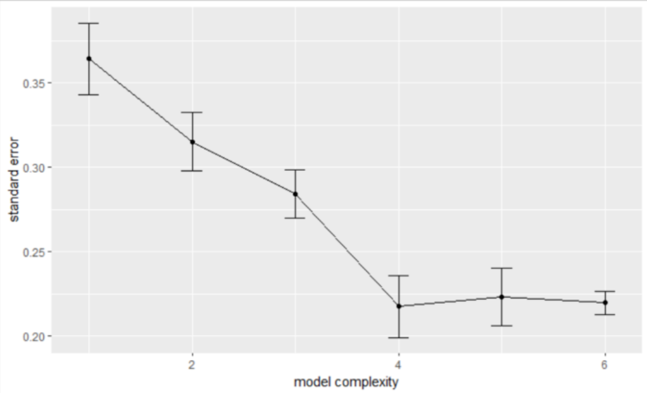
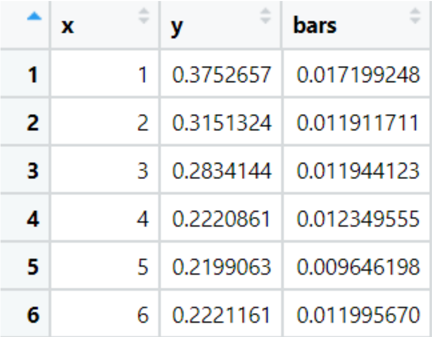


Fig 2: Plot Consisting of Error Bars Which is Spread Across Six Models with second randomised test subset

<u>Cross-validation for Classification model attempt 3:</u>



	x	y	bars
1	1	0.3752657	0.017199248
2	2	0.3151324	0.011911711
3	3	0.2834144	0.011944123
4	4	0.2220861	0.012349555
5	5	0.2199063	0.009646198
6	6	0.2221161	0.011995670

Here x describes the model number, y depicts the average square error and bars depict the variance. The graph below shows the average square error rate and model complexity. The error bars are used to depict the error range. Every model has a variance range which extends as a vertical bar. Slight changes are visible in the error bars from the previous attempt. However, still, model 4 has the smallest standard error of 21%. Also, model 5 and 6 still falls within the variance of the fourth model but with a comparative greater average square error rate of 22% respectively. Therefore, still, model 4 seems to be the best fit for the classification.

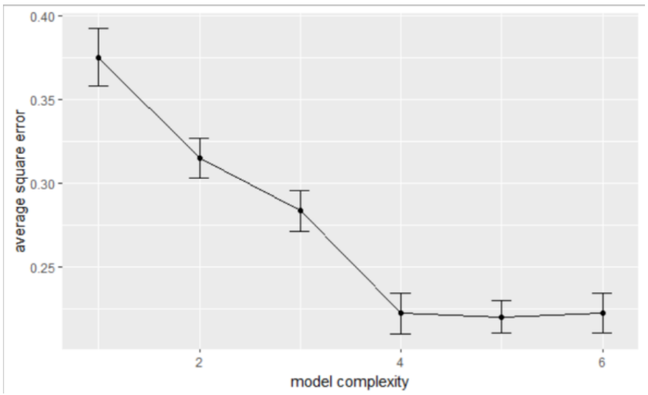
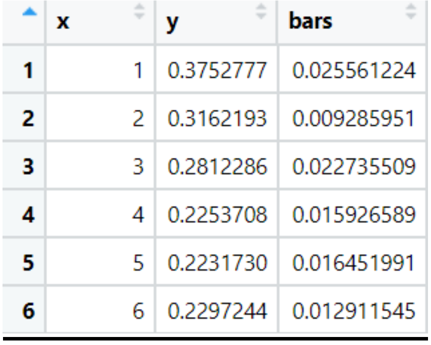


Fig 3: Plot Consisting of Error Bars Which is Spread Across Six Models with third randomised test subset

Cross-validation for Classification model attempt 4:



	x	y	bars
1	1	0.3752777	0.025561224
2	2	0.3162193	0.009285951
3	3	0.2812286	0.022735509
4	4	0.2253708	0.015926589
5	5	0.2231730	0.016451991
6	6	0.2297244	0.012911545

Here x describes the model number, y depicts the average square error and bars depict the variance. The graph below shows the average square error rate and model complexity. The error bars are used to depict the error range. Every model has a variance range which extends as a vertical bar. It is clear from the values and the plot that models four, five and six have an average square error rate of 21% respectively. Therefore, still, model 4 seems to be the best model.

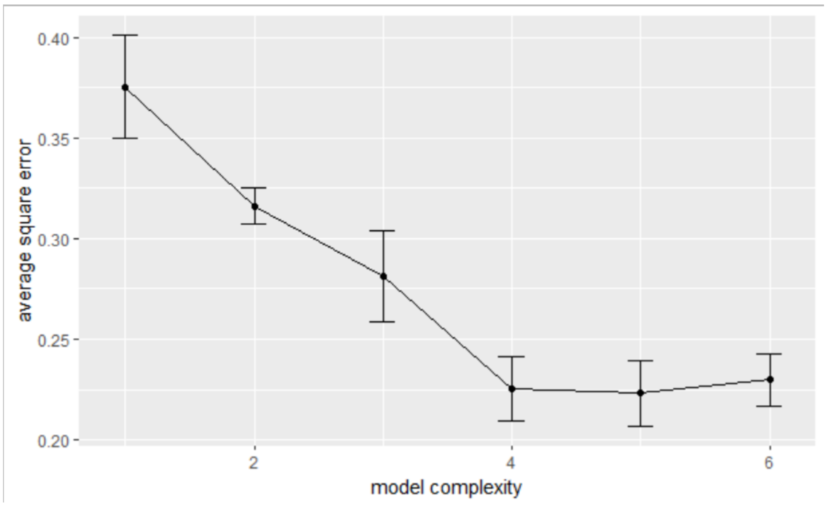
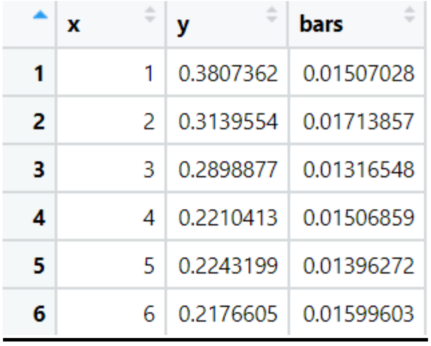


Fig 4: Plot Consisting of Error Bars Which is Spread Across Six Models with fourth randomised test subset

Cross-validation for Classification model attempt 5:



▲	x	y	bars
1	1	0.3807362	0.01507028
2	2	0.3139554	0.01713857
3	3	0.2898877	0.01316548
4	4	0.2210413	0.01506859
5	5	0.2243199	0.01396272
6	6	0.2176605	0.01599603

Here x describes the model number, y depicts the average square error and bars depict the variance. The graph below shows the average square error rate and model complexity. The error bars are used to depict the error range. Every model has a variance range which extends as a vertical bar. It is clear from the values and the plot that models four, and five have an average square error rate of 21.9% respectively. However, model 4 has a lower variance than that of model 5. Therefore, still, model 4 seems to be the best model.

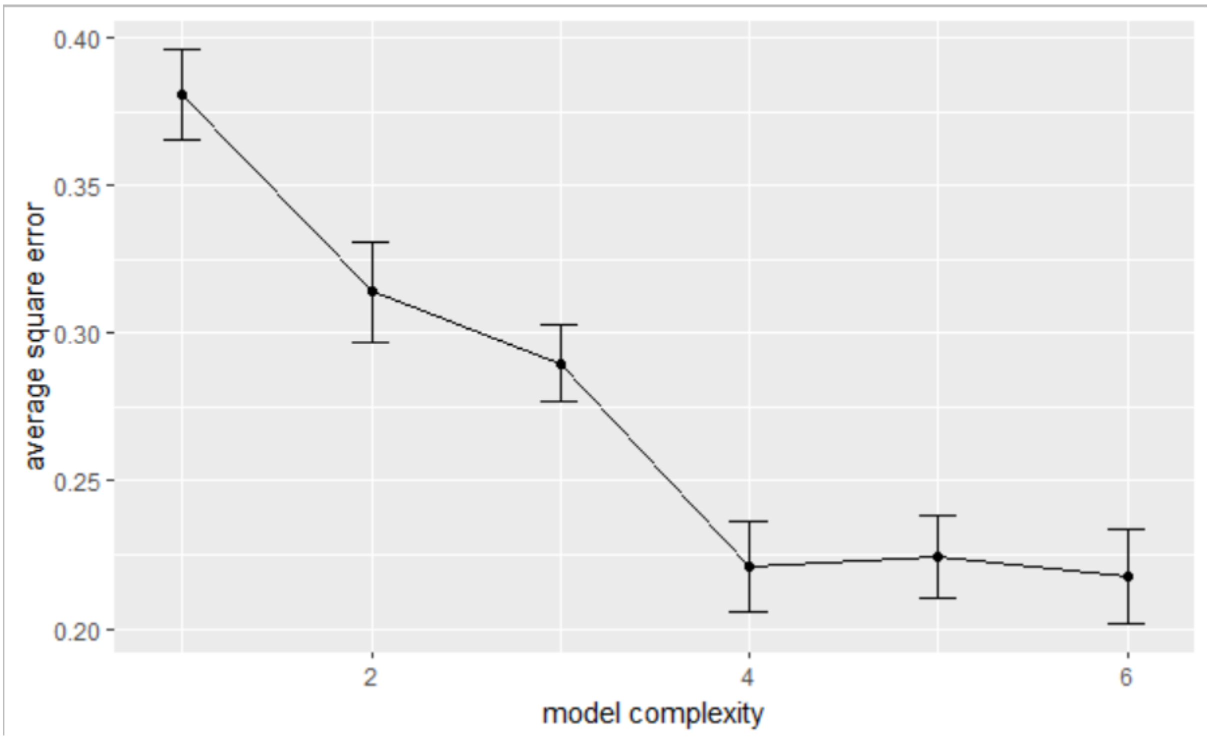


Fig 5: Plot Consisting of Error Bars Which is Spread Across Six Models with fifth randomised test subset

REGRESSION:

The relationship between two or more variables in a dataset is described using a regression model. The regression model is built using the "Spanish Wine" dataset. The wine rating is treated as a dependent variable, while the other variables are treated as independent variables that can be utilised to determine the link between them. Furthermore, either linear or logarithmic regression is used in the regression model. Although the Spanish wine dataset and several regression models have been thoroughly addressed, I will explain cross-validation approaches that can be utilised to validate the regression models created in this report. The regression model is validated using the 5-fold cross-validation procedure. A linear regression model is created to establish the relationship between rating as a function of the year, number of reviews, body, acidity and price. Here the price is considered a polynomial function.

The dataset is cleaned and resampled into the test and train dataset as the value of k is determined to be 5. Several iterations were performed to identify the best model for the regression model, and the best model was identified.

Cross-validation for Regression model attempt 1:

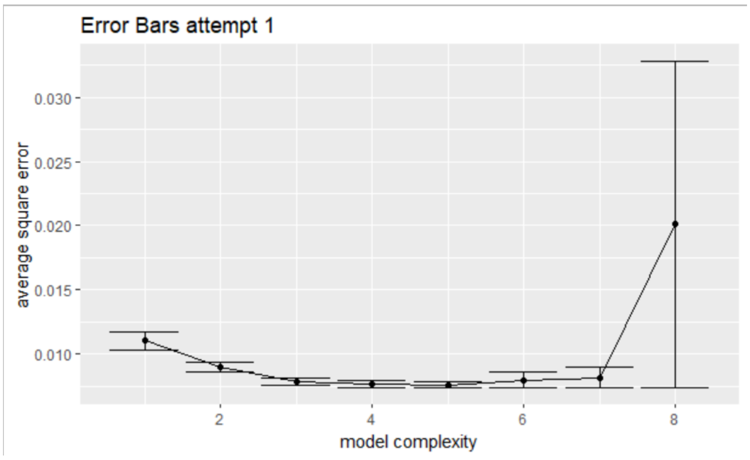


Fig 6: model complexity vs average square error for a regression model with first fold test data

The dataset is typically split in a 4:1 ratio, with 4 data folds being the training dataset and 1 data fold termed the test dataset. The regression model is performed, and the error graph is presented as a result. The average square errors are plotted, as well as the error variance that corresponds to them. The error bars are used to keep the variance limitations in check. The graph shows that as the model complexity rises, the average square error decreases. To put it another way, when the degree of price level rises, the model tends to overfit, lowering the average square error. Any model after 4 tends to overfit the data, as shown in the graph. Model 3 is the best model according to the first try because it is within the variance range of model 4.

Cross-validation for Regression model attempt 2:

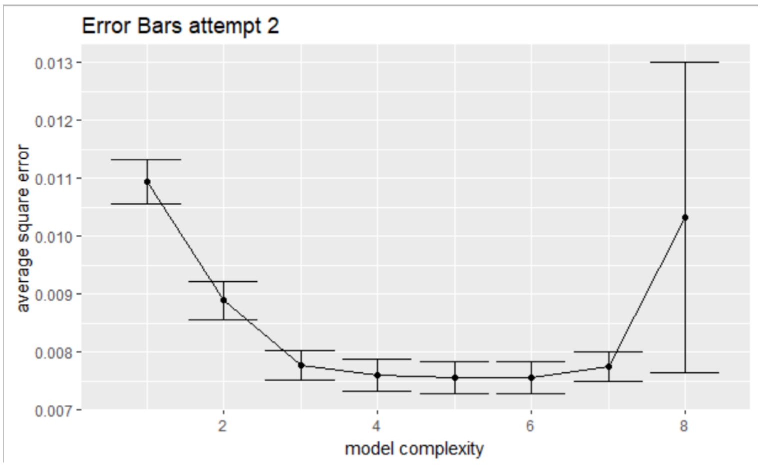


Fig 7: model complexity vs average square error for a regression model with second fold test data

Here the second fold test dataset is used to predict the output of the regression model. The average square errors are plotted, as well as the error variance that corresponds to them. The error bars are used to keep the variance limitations in check. For all models, the variance range is noticeable. Furthermore, model 8 shows a significant deviation, indicating that it is overfitting the data. Nonetheless, model 5 has the lowest average square error rate; however, the square errors of models 3 and 5 are just a little different. As a result, this tiny change can be overlooked, and Model 3 can still be selected because it is less complicated.

Cross-validation for Regression model attempt 3:

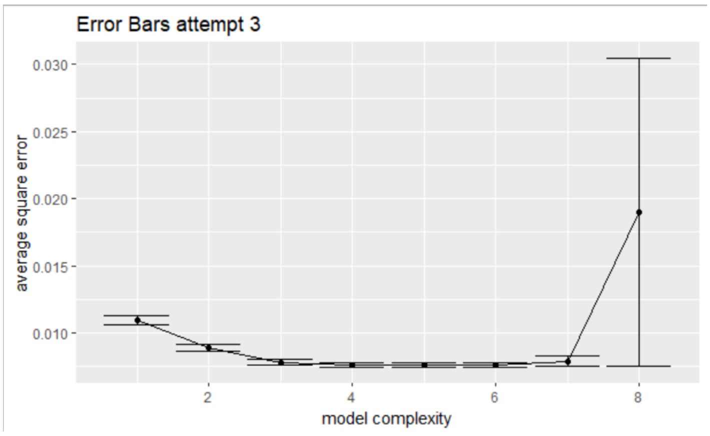


Fig 8: model complexity vs average square error for a regression model with third fold test data

Here the third fold test dataset is used to predict the output of the regression model. The average square errors are plotted, as well as the error variance that corresponds to them. The error bars are used to keep the variance limitations in check. The variance of the eighth model covers all the remaining models' average square errors. As a result, model 8 overfits the data. Furthermore, any model with a degree greater than 4 is overfitting. Model 3 is still regarded as the best model.

Cross-validation for Regression model attempt 4:

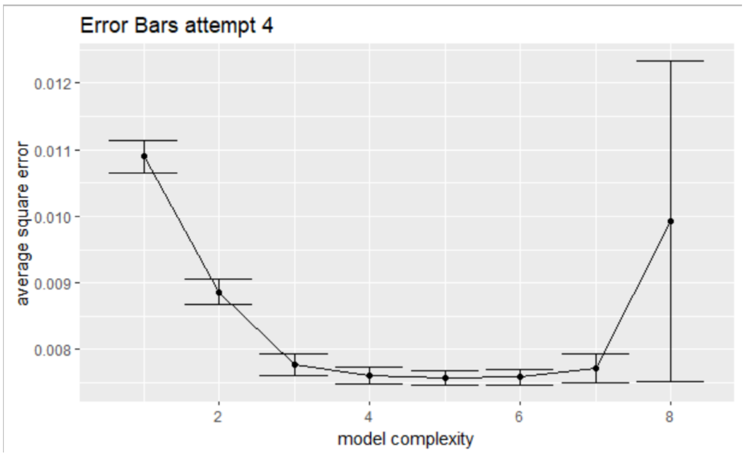


Fig 9: model complexity vs average square error for a regression model with fourth fold test data

The regression model's output is predicted using the fourth fold test dataset. Model 5 has the smallest average square error. Model 3's variance, on the other hand, is still within the bounds of model 5's variance. As a result, model 3 is the best model according to the one standard error rule.

Cross-validation for Regression model attempt 5:

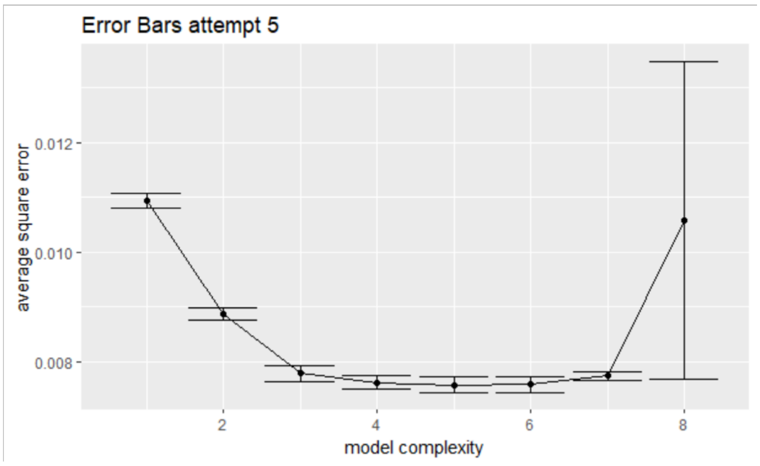


Fig 10: model complexity vs average square error for a regression model with fifth fold test data

To anticipate the regression model's output, the fifth fold test dataset is employed. Model 5 has the lowest average square error. Model 3's variance, on the other hand, remains within the bounds of model 5's variance. As a result, model 3 is regarded the best model based on the one standard error rule.

CONCLUSION:

The cross-validation techniques are used in the regression and classification model and the corresponding output is obtained. Model 4 seems to be the best according to the output obtained for the classification model and model 3 seems to be the best for the regression model. The optimal model is selected using the one standard error rule.

REFERENCE:

1. Curated dataset from Brightspace: https://dal.brightspace.com/d2l/le/content/221958/viewContent/3012797/View
2. fedesoriano. (September 2021). Heart Failure Prediction Dataset. Retrieved [Date Retrieved] from https://www.kaggle.com/fedesoriano/heart-failure-prediction.
3. Professor Scott Flemming code on cross-validation for classification. – "chapter 5 exercise 2 - CV using Classification (LDA and the Iris data set) and the 1SE rule"

R-CODE:

Classification model:

```
library(tidyverse)
library(dplyr)
library(MASS)
library(ggplot2)
library(readr)
library(corrplot)


#importing dataset
setwd("C:/CANADA/industrial engineering/summer term/data analytics/dataset")
heartdata= read.csv("heart.csv")
head(heartdata)
str(heartdata)
dim(heartdata)


#cleaning dataset
colSums(is.na(heartdata))
cleandata <- heartdata[complete.cases(heartdata),]
dim(cleandata)
```

```
view(cleandata)
str(cleandata)
cleandata$HeartDisease <- as.factor(cleandata$HeartDisease)

view(cleandata)
heartMix = slice(cleandata, sample(1:n()))
View(heartMix)
summary(cleandata)
summary(heartMix)
id = seq(1, 918, by=1)
heartRando = mutate(heartMix, id)
view(heartRando)

k = 5
numRows = nrow(heartRando)
errors = rep(0, k)
totalError = 0
id = seq(1, 918, by=1)
heartMix = slice(cleandata, sample(1:n()))
heartRando = mutate(heartMix, id)
#number of models I want to try and their definitions
numModels = 6
modelnames = rep(0, numModels)
modelnames[1] = "HeartDisease~Cholesterol+RestingBP+FastingBS"
modelnames[2] = "HeartDisease~ Age+MaxHR"
modelnames[3] = "HeartDisease~ Age+MaxHR+FastingBS"
modelnames[4] = "HeartDisease~ Age+MaxHR+FastingBS+ChestPainType"
modelnames[5] = "HeartDisease~ Age+MaxHR+FastingBS+ChestPainType+RestingECG"
modelnames[6] = "HeartDisease~Age+Sex+Oldpeak+ExerciseAngina"
model = lda(eval(parse(text=paste(modelnames[j])), train))
```

```
model
heartMix = slice(cleandata, sample(1:n()))
heartRando = mutate(heartMix, id)
errors = matrix(1:numModels*k,
                dimnames= list(seq(1, numModels, by=1), seq(1, k, by=1)),
                nrow=numModels, ncol=k)
print(errors)
class(errors)
view(numModels)
k
avgError = rep(0,numModels)
for(j in 1: numModels){
  for(i in 1:k){

  }
}

for(j in 1:numModels){
  k = 5
  numRows = 918
  totalErrors = rep(0, k)
  for(i in 1:k){
    test = filter(heartRando, id >= (i-1)*numRows/k+1 & id <= i*numRows/k)
    train = anti_join(heartRando, test, by="id")
    model = lda(eval(parse(text=paste(modelnames[j])), train))
    modelGuesses = predict(model, test)
    errors[j,i] = 1-mean(modelGuesses$class == test$HeartDisease)
    totalErrors[i] = errors[j,i]+totalErrors[i]
  }
  avgError[j] = totalErrors[j]/k
}

avgError

view(totalErrors)

mean(errors[1,])

mean(errors[2,])

mean(errors[3,])

mean(errors[4,])

mean(errors[5,])

mean(errors[6,])

plot(c(mean(errors[1,]), mean(errors[2,]), mean(errors[3,]), mean(errors[4,]),
mean(errors[5,]), mean(errors[6,])))

sqrt(var(errors[1,])/k)

x = seq(1, numModels, by=1)

y = rep(0, numModels)

bars = rep(0, numModels)

for(i in 1:numModels){

  y[i] = mean(errors[i, ])

  bars[i] = sqrt(var(errors[i,])/k)

}

allData = data.frame(x, y, bars)

View(allData)

ggplot(allData, aes(x=x, y=y)) +

  geom_line() +

  geom_point()+

  geom_errorbar(aes(ymin=y-bars, ymax=y+bars), width=.2,

                position=position_dodge(0.05))+

  xlab("model complexity")+

  ylab("average square error")
```

<u>Regression model:</u>

```
library(tidyverse)

library(dplyr)
library(scales)
library(ggplot2)
library(leaps)

#setting up working directory
setwd("C:/CANADA/industrial engineering/summer term/data analytics/dataset")
#importing the dataset
dataset= read.csv("spanish_wine.csv")
view(dataset)
dim(dataset)
str(dataset)
summary(dataset)

colSums(is.na(dataset))
cleandata <- dataset[complete.cases(dataset),]
dim(cleandata)
cleandata$year <- as.integer(cleandata$year)

view(cleandata)
#cleaning dataset
New_wine = drop_na(cleandata)
view(New_wine)
str(New_wine)
dim(New_wine)

attach(New_wine)
#model3
wine= dplyr::select(New_wine, year,rating, num_reviews, body, acidity, price)
for (i in 8) {
  best3 = regsubsets(rating~year+num_reviews+body+acidity+poly(price,i, raw = TRUE),data= wine,nvmax = 12)
  summary(best3)
  summary(best3)$rsq
  plot(summary(best3)$rsq, type = "o",xlab = "MODEL",ylab = "R squared value")
}

library(boot)
cross_validation_error = rep(0,8)
view(cross_validation_error)
for(i in 1:8){
  model=glm(rating~year+num_reviews+body+acidity+poly(price,i, raw = TRUE))
  cross_validation_error[i] = cv.glm(wine, model, K=5)$delta[1]
}
cross_validation_error
view(cross_validation_error)

x=seq(1,8, by=1)
cv = data.frame(x, y = cv.error)
cv
ggplot(data=cv)+geom_point(aes(x=x, y=y))+
  xlab("Model Complexity")+
  ylab("Model Error")



attach(wine)
k=5
adjustedrsqds = rep(0,8)
adjustedrsqds
rsqVals =rep(0,8)
rsqVals
for(k in 1:8){
  model = lm(rating~year+num_reviews+body+acidity+poly(price,k, raw = TRUE))
  adjustedrsqds[k] = summary(model)$adj.r.squared
  rsqVals[k] = summary(model)$r.squared
}
adjustedrsqds
rsqVals
plot(adjustedrsqds)
plot(rsqVals)

#K-FOLD validation

k.fold.errors.10 = rep (0 ,8)
for(k in 1:10){
  model = glm(rating~year+num_reviews+body+acidity+poly(price,i, raw = TRUE))
  k.fold.errors.10[k] = cv.glm(wine, model, K=5)$delta[1]
}
k.fold.errors.10



ModelComplexity = seq(1, 8, by=1)
various_Errors = data.frame(ModelComplexity, k.fold.errors.10, cross_validation_error,rsqVals)
View(various_Errors)

ggplot(data = various_Errors)+
  geom_point(aes(x=ModelComplexity, y=k.fold.errors.10), col="red")+
  geom_point(aes(x=ModelComplexity, y=cross_validation_error), col="blue")

ggplot(data = various_Errors)+
  geom_point(aes(x=ModelComplexity, y=adjustedrsqds), col="green")+
  geom_point(aes(x=ModelComplexity, y=rsqVals), col="yellow")+
  geom_path(aes(x=ModelComplexity, y=adjustedrsqds), col="green")+
  geom_path(aes(x=ModelComplexity, y=rsqVals), col="yellow")

attach(wine)
numRows = nrow(wine)
id = seq(1, numRows, by =1)
wineShuffle = slice(wine, sample(1:n()))
wineShuffle = mutate(wineShuffle, id)
View(wineShuffle)
k = 5

errors = matrix( nrow = 8, ncol = 5)
View(errors)
errors[1,2] = 0
View(errors)
for(j in 1:8){
  for(i in 1:5){
    errors[j,i] = 0
  }
}
View(errors)

library(dplyr)
library(MASS)
library(ggplot2)
totalError = 0
for(j in 1:8){
  for(i in 1:k){
    test= filter(wineShuffle, id >= (i-1)*numRows/k+1 & id <=i*numRows/k)
    train = anti_join(wineShuffle, test, by="id")
    model = lm(rating~year+num_reviews+body+acidity+poly(price,j, raw = TRUE), data = train)
    errors[j,i] = mean((test$rating - predict.lm(model, test))^2)
  }}
View(errors)


avgRegEr = rep(0,8)
avgRegEr
for(j in 1:8){
  for(i in 1:5){
    avgRegEr[j] = avgRegEr[j]+errors[j, i]
  }
}
avgRegEr
avgRegEr/k
cross_validation_error


se = rep(0, 8)
se
for (i in 1:8){
  se[i] = sqrt(var(errors[i,])/k)
}
se


x = seq(1,8, by = 1)
wineBest = data.frame(x,avgRegEr/k , se)
wineBest


library(ggplot2)
ggplot(data = wineBest, aes(x = x, y=avgRegEr.k))+
  geom_point()+
  geom_line()+
  geom_errorbar(aes(ymin = avgRegEr.k-se, ymax = avgRegEr.k +se))+
  xlab("model complexity")+
  ylab("average square error")+
  ggtitle("Error Bars attempt 5")
```